Numerical Simulations For Active Tectonic Processes: Increasing Interoperability And Performance

Interoperability Milestone J due 2/27/2004

Full implementation using improved codes. Review board approves integration into completed framework and updated documentation for:

- Mesh generation - Demonstrate adaptive mesh capability within GeoFEST using a fault stepover geometry wherein the mesh is adapted to accommodate large strains in the stepover as the displacement on the main faults grows.
- Virtual California
- SLIDER
- Phase Dynamical Probability Change Index
- Data Mining via Karhunen-Loeve Space-Time Pattern Analysis
- STRESSCO* codes (as an example of FLTGRV, FLTGRH, STRGRV, STRGRH)
- DataMining via Genetic Algorithm Analysis
- Hidden Markov Model - demonstrate interaction with federated DB through framework
- Disloc
- Visco**
- SIMPLEX***
- Final fault DB for California with documentation (now called QuakeTables)

**STRESSCO is the initialization code for Virtual California. There is no direct user interaction with this code, and it is now integrated into Virtual California.*

***Visco is now known as GeoFEST*

****SIMPLEX plus GMT Visualization is now known as GeoFit*

JPL Clearance: CL#04-1980

Team

Andrea Donnellan:
Principal Investigator
Jet Propulsion Laboratory
Mail Stop 183-335
4800 Oak Grove Drive
Pasadena, CA 91109-8099
donnellan@jpl.nasa.gov
818-354-4737

Geoffrey Fox:
Information Architect

Community Grid Computing Laboratory
Indiana University
501 N. Morton, Suite 224
Bloomington, IN 47404-3730
gcf@indiana.edu
812-856-7977

Terry Tullis:
Fast Multipole Methods
Brown University
Box 1846, Brown University
Providence, RI 02912-1846
Terry_Tullis@Brown.edu
401-863-3829

John Rundle:
Pattern Recognizers
Center for Computational Science and Engineering
U. C. Davis
Davis, CA 95616
rundle@geology.ucdavis.edu
530-752-6416

Greg Lyzenga:
Inversion and Assimilation Codes

Jet Propulsion Laboratory
Mail Stop 126-347
4800 Oak Grove Drive
Pasadena, CA 91109-8099
gerg.lyzenga@jpl.nasa.gov
818-354-6920

Jay Parker:
Overall Software Engineer
Jet Propulsion Laboratory
Mail Stop 238-600
4800 Oak Grove Drive
Pasadena, CA 91109-8099
Jay.W.Parker@jpl.nasa.gov
818-354-6790

Marlon Pierce:
Code Interoperability Software Engineer

Community Grid Computing Lab
Indiana University
501 N. Morton, Suite 224
Bloomington, IN 47404-3730
marpierc@indiana.edu
812-856-1212

Dennis McLeod:
Database Interoperability
Professor
Computer Science Department
University of Southern California
Los Angeles, CA 90089-0781
mcleod@usc.edu
213-740-4504

Lisa Grant:
Fault Database Architect
University of California, Irvine
Environmental Analysis and Design
Irvine, CA 92697-7070
lgrant@uci.edu
949-824-5491

Robert Granat:
Pattern Recognizers

Jet Propulsion Laboratory
Mail Stop 126-347
4800 Oak Grove Drive
Pasadena, CA 91109-8099
robert.granat@jpl.nasa.gov
818-393-5353

Executive Summary

Background

The QuakeSim Problem Solving Environment has been developed according to the design document posted at:

http://quakesim.jpl.nasa.gov/DesignDocument2.1.pdf.

This milestone report documents the implementation of a system using this interoperable framework for linking processes on diverse remote host machines. We will describe posted supporting documents, the status of the QuakeSim Gateway framework, and the resources and processes that have been integrated with this framework.

Current Status

We have implemented the Gateway Problem Solving Environment framework (QuakeSim) and integrated the required Milestone J geophysical applications into the framework.

Progress Since Interoperability Milestone I

- Added all of the milestone codes: developed user interfaces, made use of available services (particularly context management, file transfer, and Apache Ant services for job management).
- Added multiple user support. The previous portal only supported a single user account.
- Wrapped GMT and IDL services for visualization.
- Developed GPS and Seismic catalog databases and services, integrating them into the portal, and integrating them with RDAHMM.
- Developed ssh wrappers to remotely execute file transfers and command execution through the portal.
- Improved packaging of the portal using Apache Ant for compiling and deploying.
- Developed initial testing framework using HTTPUnit.

Supporting Documents

The project software engineering plan is located at:

http://www-aig.jpl.nasa.gov/public/dus/gem/CT/SW_Eng_plan.html

Requirements for the project have been defined and can be found at:

http://quakesim.jpl.nasa.gov/CT_Requirements.doc

Documents specific to this milestone:

The updated QuakeSim Portal **Installation Guide** is posted at:

http://quakesim.jpl.nasa.gov/PortalInstallationGuide.pdf

The updated QuakeSim Portal **User's Guide** is posted at:

http://quakesim.jpl.nasa.gov/PortalUserGuide.pdf

The updated QuakeSim Portal User **Validation document** is posted at:

http://quakesim.jpl.nasa.gov/PortalUserValidation.pdf

The updated QuakeSim Portal **Software Test Plan** is posted at:

http://quakesim.jpl.nasa.gov/PortalSoftwareTestPlan.pdf

The updated QuakeSim Portal **Requirement Traceability Matrix** is posted at:

http://quakesim.jpl.nasa.gov/PortalRequirementTraceabilityMatrix.pdf

The fault database, now named **QuakeTables** is described here:

http://quakesim.jpl.nasa.gov/QuakeTables.doc

QuakeSim Portal Framework

We have implemented the Gateway Problem Solving Environment framework (QuakeSim) and integrated the required 12 geophysical applications into it.

QuakeSim will be used by the seismological, crustal deformation, and tectonics communities for developing an understanding of active tectonic and earthquake processes. In the figure below, the solid earth research virtual observatory (SERVO) shows science users interacting with interface programs as well as modeling, simulation, and analysis tools. The general architecture follows the "Web Services" model being developed by business interests, but is applied to scientific applications and supporting software resources (such as databases).

The system is divided into three tiers: a user interface layer (implemented as a browser interface), a system resource layer, and a middle control layer that maintains proxies (or brokers) to the system resources (Figure 1). The middle tier provides a uniform interface to the resource layer. Following the Web Services approach, we define XML interface abstractions (in WSDL) for basic services (such as File Management) and implement the interface with appropriate technologies (such as with a relational database). Communication between the services is done with an XML messaging architecture (SOAP).

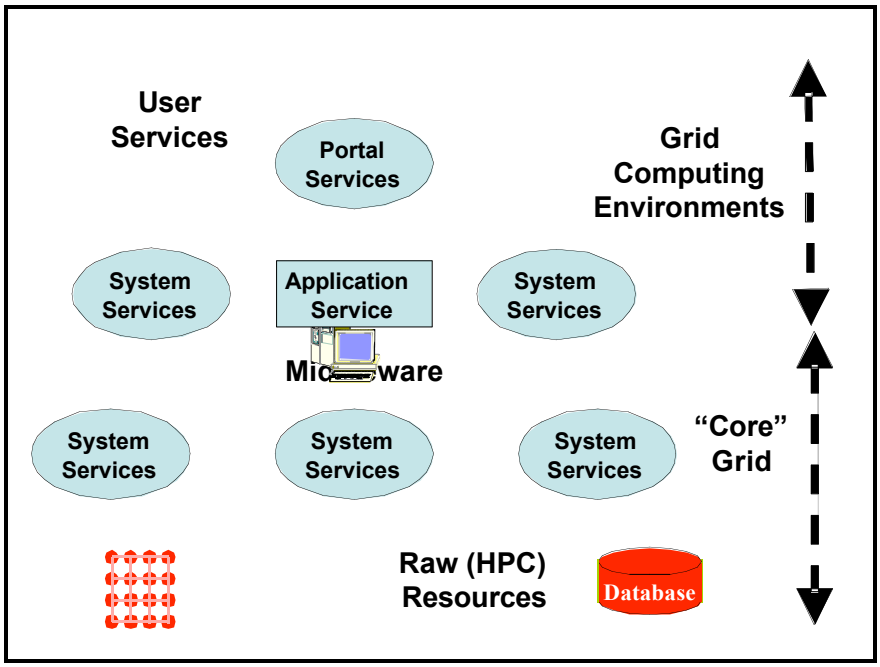

Figure 1. High level architecture of planned system showing grids, portals, and grid computing environments.

One of the most critical aspects of our implemented system is supporting interoperability given the heterogeneous nature of data sources as well as the variety of application programs, tools, and simulation packages that must operate with data from our system. Interoperability has been implemented by using distributed object technology combined with development of object API's that conform to emerging standards. We defined our object API's in XML and dynamically mapped this specification into the chosen object model.

The implementation of this framework is an important step toward our goal of a community modeling environment that includes:

1. A database system for handling both real and simulated data.
2. Fully three-dimensional finite element code with adaptive mesh generator capable of running on workstations and supercomputers for carrying out earthquake simulations.
3. Inversion algorithms and assimilation codes for constraining the models and simulations with data.
4. A collaborative portal (Object Grid Framework) allowing for seamless communication between codes, reference models, and data.

5. Pattern recognizers capable of running on workstations and supercomputers for analyzing data and simulations.

Portal Status

Current Status

The QuakeSim portal framework is complete and has integrated the required Milestone J applications.

Progress (since the last Interoperability milestone)

- Added all of the milestone codes: developed user interfaces, made use of available services (particularly context management, file transfer, and Apache Ant services for job management).
- Added multiple user support. The previous portal only supported a single user account.
- Wrapped GMT and IDL services for visualization.
- Developed GPS and Seismic catalog databases and services, integrating them into the portal, and integrating them with RDAHMM.
- Developed ssh wrappers to remotely execute file transfers and command execution through the portal.
- Improved packaging of the portal using Apache Ant for compiling and deploying.
- Developed initial testing framework using HTTPUnit.
- Portal interface improvements based upon user feedback.

Background

The QuakeSim portal and test bed consists of the following pieces:

1. A portlet-based portal environment that allows users to customize their displays and service interfaces.
2. Support for the QuakeSim codes Disloc, Simplex/Geofit, GeoFEST, VirtualCalifornia, Mesh Generation (Apollo/Akira), SLIDER, Phase Dynamical Probability (PDPC), Karhunen-Loeve Space-Time Pattern Analysis, Genetic Algorithm Analysis codes, and Hidden Markov Model Codes (RDAHMM).
3. XML based Web services that allow the user to
    - Transfer files from desktop to backend and between backend resources

- Submit jobs
- Monitor job progress on backend resources
- Archive and resubmit old jobs
- Visualize outputs
- Download output data
- Provide both human and application access to various databases
- Manage portal applications.

The general architecture is shown in Figure 2. For the test bed, we use the following resources:

- Complexity, a Sun Sunfire 880 server, acts as the main web server, runs the portal, and manages connections to remote services through client stubs.
- The fault database is hosted at USC.
- Services for job submission, file management, and job monitoring are hosted on a range of Linux and Solaris servers (danube, kamet, and grids).
- Complexity also hosts several local services for session management.
- Users may access other resources through SSH wrappers.

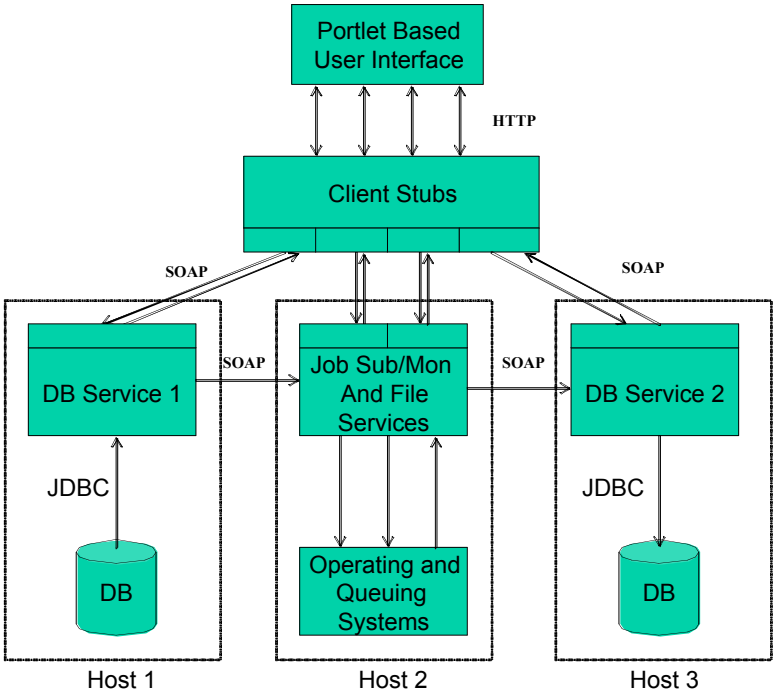

Figure 2. QuakeSim Architecture

The portal is based primarily on the following technologies:

- Jetspeed is used to build the portal framework (for which we have added some extensions).
- JavaServer Pages are used to create the web pages.
- Remote invocations and communications between various machines and databases in the testbed use Web services running in the Apache Axis implementation of WSDL and SOAP.
- We use apache-ant (with various custom extensions) to coordinate related tasks (such as the steps needed to create the initial mesh or to generate the movie).

Fault database - QuakeTables

The fault database has been generalized since last year, now being termed QuakeTables. We have augmented and cleaned up the user interface, adding more guiding information for the inexperienced. We also added data, and some new data parameters for the California faults data.

This includes data from the California Geological Survey database. We have provided specifications, documentation and papers describing the system from the high level - as a framework that can be used for many kinds of geoscience data; we also provide very low level detail to support the operation of the current QuakeTables system.

A more complete description of QuakeTables may be found in:

http://quakesim.jpl.nasa.gov/QuakeTables.doc

Running Applications through the Portal

Examples of typical use of each integrated application through the portal are described in the QuakeSim Portal User Guide:

http://quakesim.jpl.nasa.gov/PortalUserGuide.pdf

Demonstrating GeoFEST with Adaptive Meshing

As required by Milestone J, these are the instructions for demonstrating the execution of GeoFEST with adaptive meshing through the QuakeSim portal. The demonstration consists of running a parallelized version of GeoFEST installed on losangeles.jpl.nasa.gov and downloading a plot of adaptively refined finite element mesh generated by GeoFEST. This is not included in the QuakeSim Portal User Guide, as a general user will not yet use it.

Requirements

1. You must acquire an account on losangeles.jpl.nasa.gov. If you do not have one, please contact Charles Norton, Charles.D.Norton@jpl.nasa.gov
2. You must have a PDF viewer such as Adobe Acrobat installed on your desktop. If you do not, you may download and install a free version of Acrobat from http://www.adobe.com/products/acrobat/readstep2.html.
3. Optionally, you may want to acquire a portal account. This is explained in the QuakeSim User Guide. If you prefer, you may instead use one of the default accounts described in the user guide.
4. You should use either Internet Explorer or Mozilla for your web browser.

Running the Demo

1. Log into the QuakeSim portal and select “GeoFEST_Adaptive” from the list of available applications.

2. Set up a problem geometry and generate an initial mesh. You may use the Northridge example as described in the QuakeSim User Manual. This sample problem is also shown in the "validation" account (username and password are validation/validation).

3. Launch GeoFEST. This is done from the screen shown in Figure 3. Pay special attention to the upper right hand corner of the screen. **You must provide the correct username and password for your losangeles account**. If you want email notification when GeoFEST completes, you should also make sure that your email address is correct.

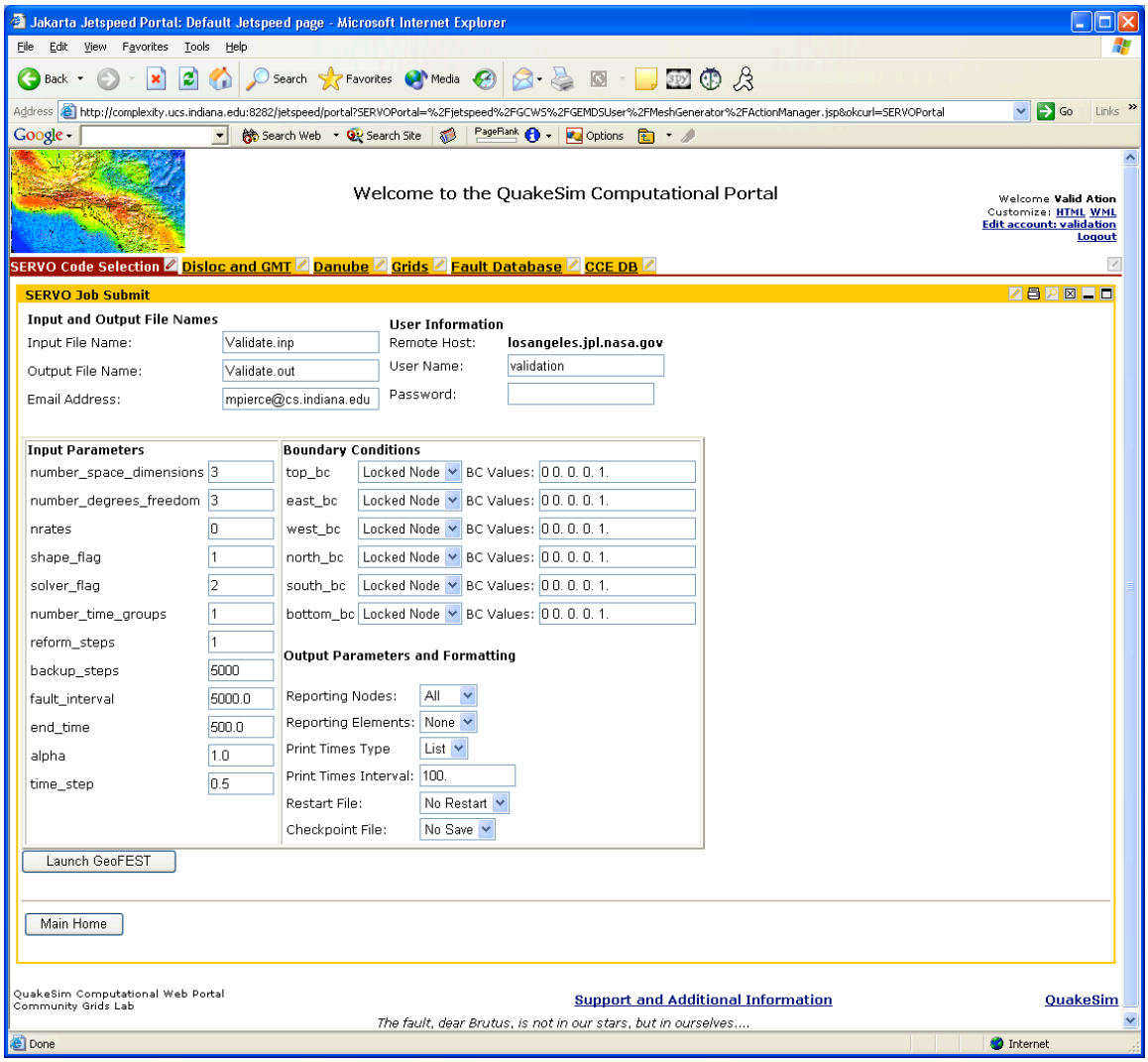

Figure 3. GeoFEST submission form for the adaptive meshing demonstration.

4. When GeoFEST has completed, you will receive an email notification. For the provided parameters, this will take a few minutes.

5. After receiving your email that GeoFEST has completed, click "GeoFEST_Adaptive" from the portal main code menu. You should see the screen shown in Figure 4.

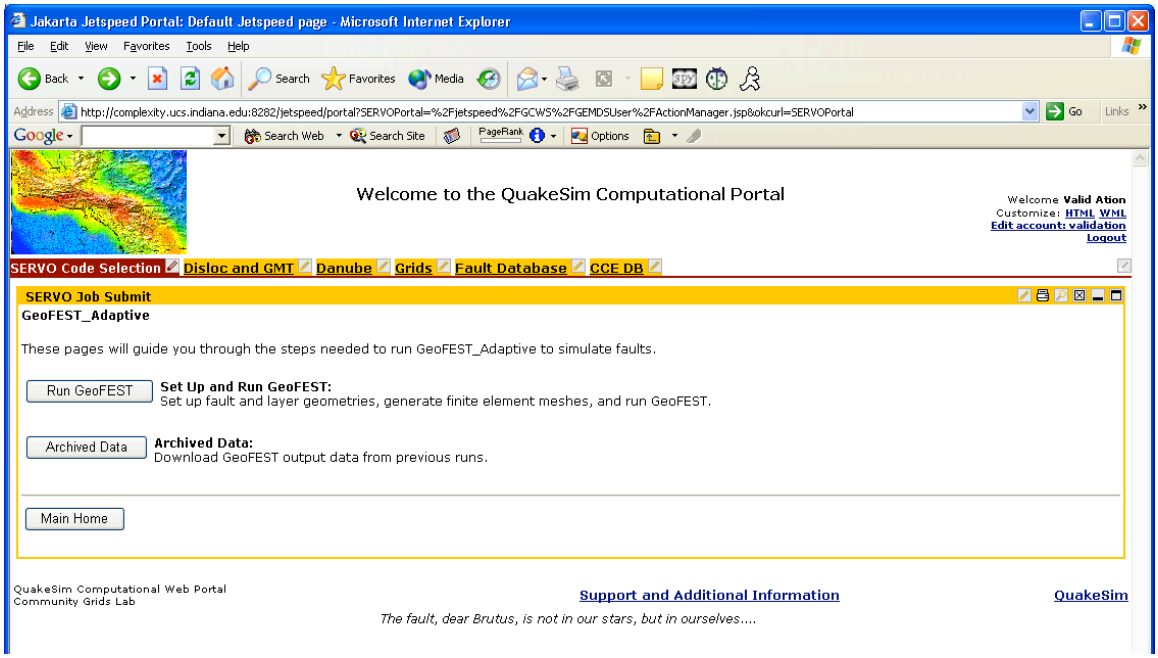

Figure 4. Viewing the output image in the archive.

6. Click "Archived Data" to view a list of archived projects and their meshes. This list will resemble the screen shown in Figure 5. Click the link under "Mesh Image".

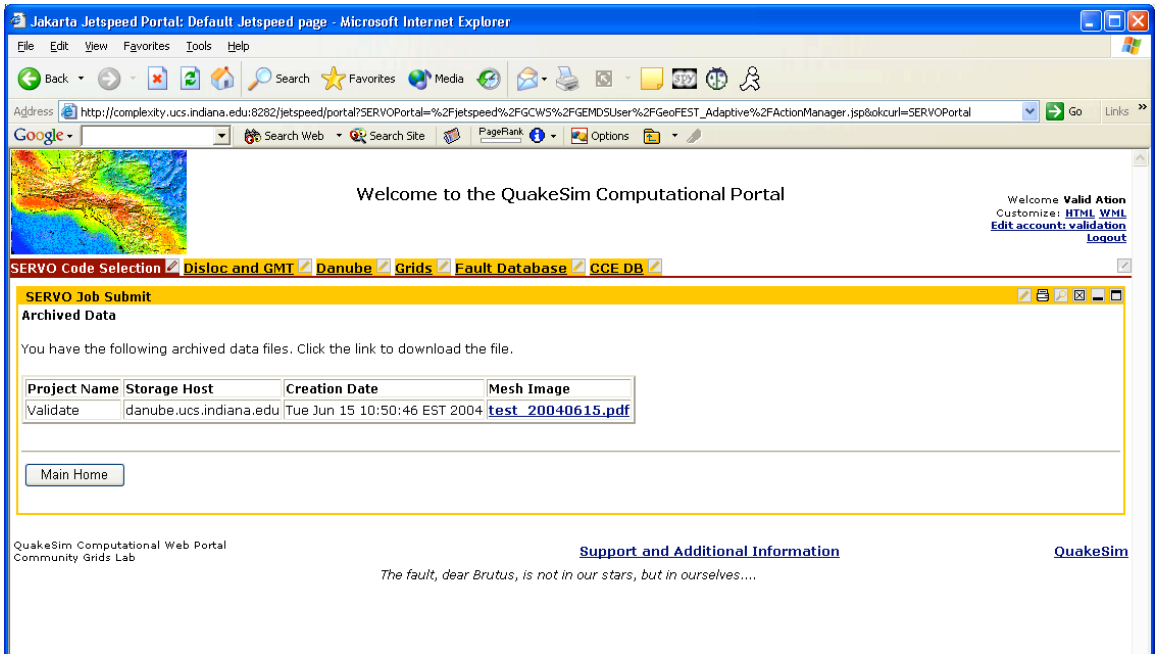

Figure 5. Select a mesh image to view.

7. Your browser may prompt you for an appropriate application for viewing the PDF image. Assuming you have installed Adobe Acrobat, use this application. You should then see a screen similar to Figure 4.

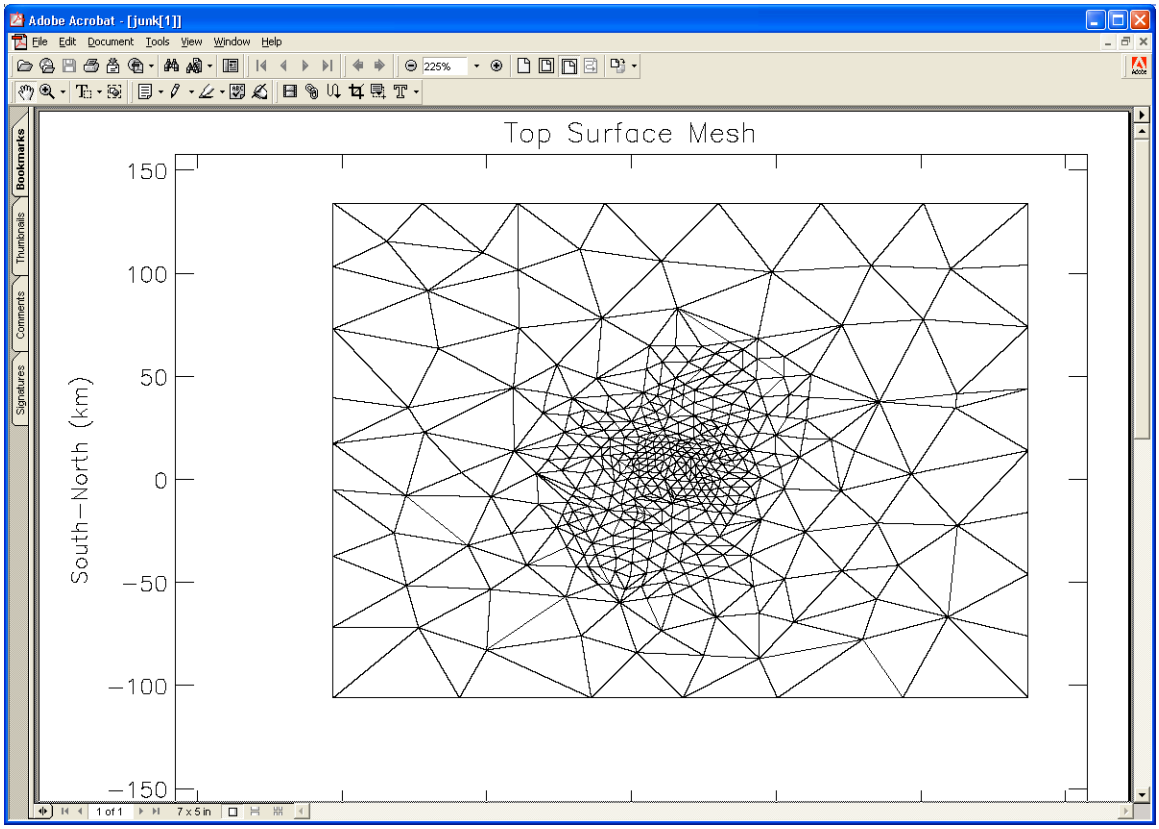

Figure 2. PDF image of the adaptive mesh used by GeoFEST.